Experience Running Embedded EPICS on NI CompactRIO

Eric Björklund

Dolores Baros

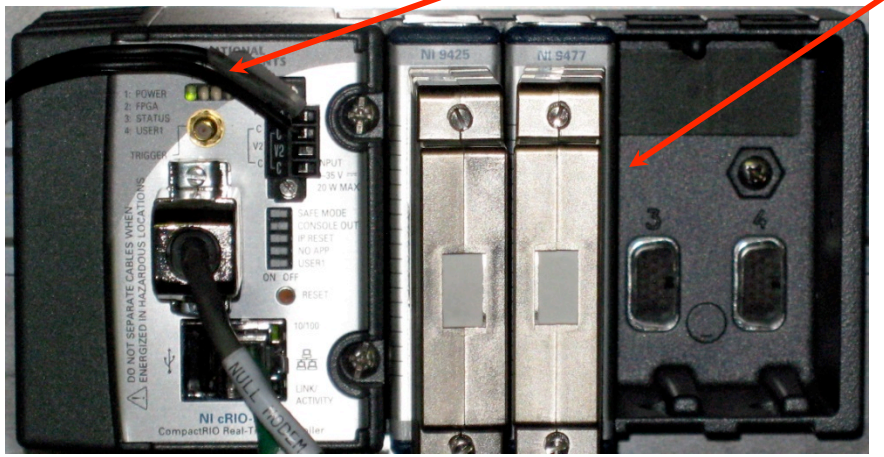
Scott Baily

Los Alamos NATIONAL LABORATORY EST.1943

LANSCE LOS ALAMOS NATIONAL LABORATORY

NNSA

What is CompactRIO?



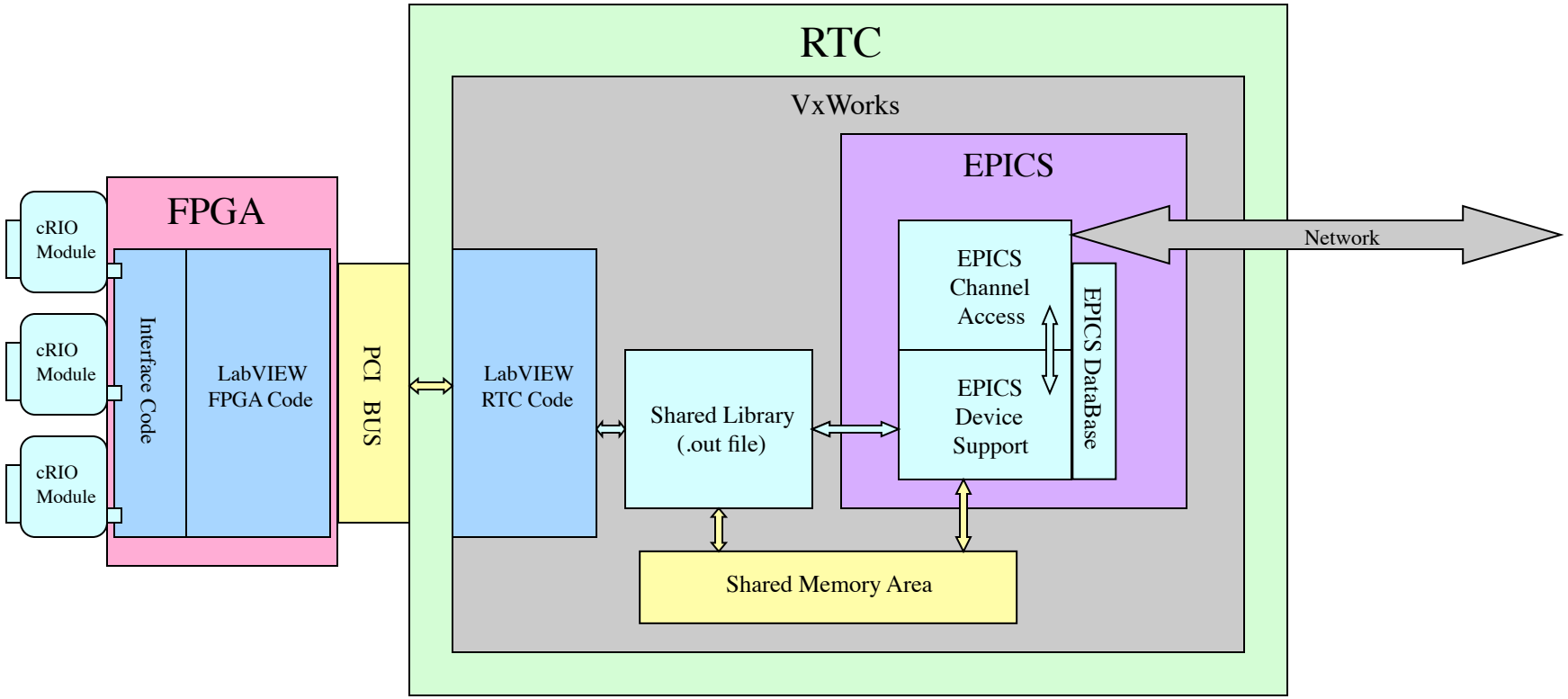
- Power-PC based Real Time Controller (RTC).
- FPGA Backplane
- Connected via PCI bus
- Designed for harsh environments

- Fast PLC
  - μSec vs. mSec Response
- Programmed in LabVIEW instead of Ladder Logic
  - Both FPGA and RTC are programmed in LabVIEW

Embedding EPICS IOC on CompactRIO

- Real Time Controller (RTC) is a Power PC running vxWorks (6.3)
- vxWorks license not required!
  - NI has permission to distribute the vxWorks header files
- Special BSP provided by NI
  - Includes NFS & Telnet
  - Priority range allocated for EPICS
  - EPICS runs at lower priority (100-199) than LabVIEW (10-100)

System Architecture



Embedded EPICS vs. NI CA Server

- Complete EPICS IOC Environment
  - All record types and fields are visible and available
  - EPICS Utilities
    - Archiver
    - Channel Access Security
    - Bumpless Reboot
    - Sequencer
- Maximum Flexibility on How the Application is Partitioned
  - FPGA / LabVIEW RTC / EPICS Database
- Not an “Out-Of-The-Box” Solution
  - Requires special BSP (available from NI on request)
  - Additional configuration required

The Project

- Replace one accelerator module's "Industrial I/O" channels with cRIO.
  - 40 binary outputs
  - 64 binary inputs
  - 32 analog inputs
  - 8 stepper motor channels
- Actually, replaced two AB Control Logix crates that had originally replaced the original 1960's "RICE" system.
  - Replaced two Control Logix crates with two cRIO crates
- Basic SCADA
  - No closed loop. No exotic timing. But….
  - System must emulate our 1960's vintage "RICE" system.

Previous Results (Vancouver, 2009)

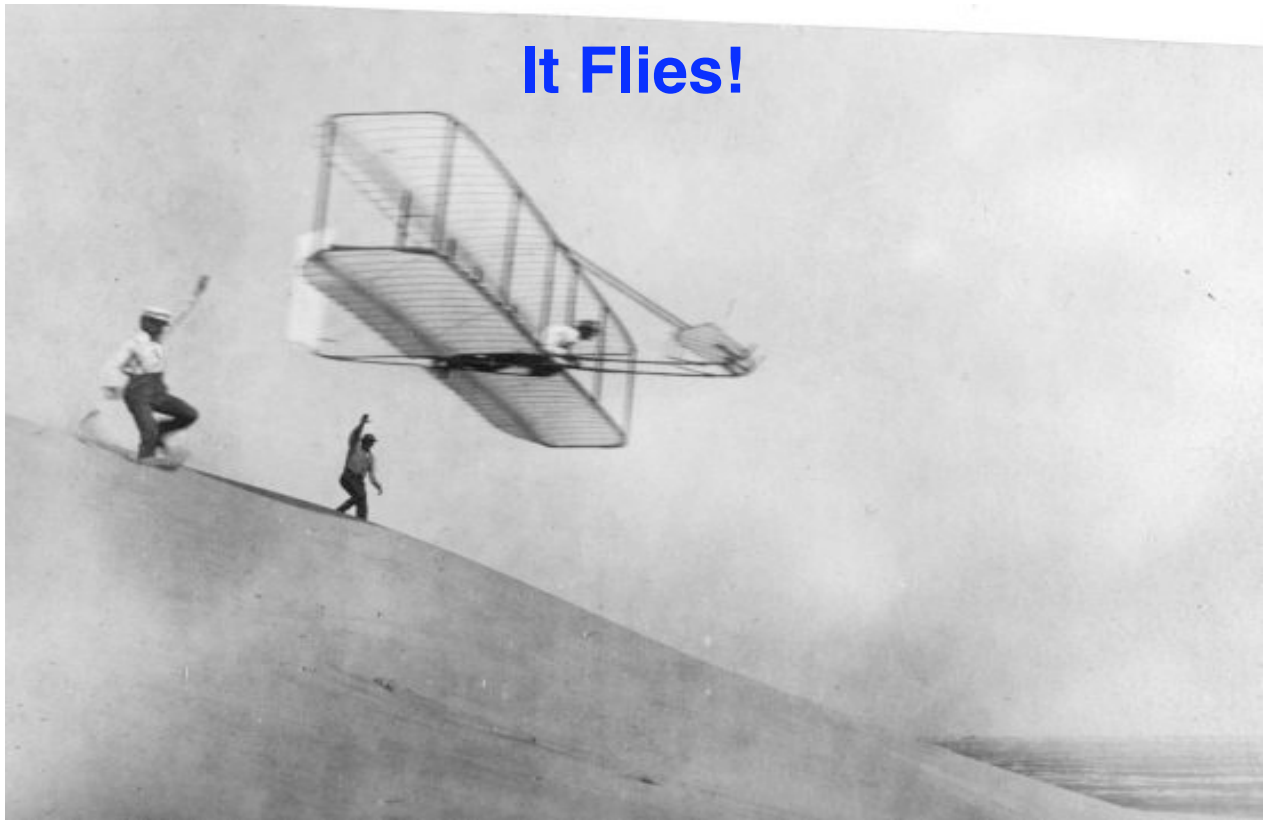
Previous Results (Vancouver, 2009)

- **Fire Alarm Forced Building Evacuation Before Results Could Be Reported.**
  - **(Nothing actually caught fire)**
- **System did work, but with limited functionality.**
  - **Only deployed the Binary I/O part during 2009 run-cycle**
  - **Our strategy of trying to push as much as possible into the FPGA resulted in running out of FPGA space (Virtex II)**
    - **Not all binary outputs could be implemented**
    - **11 hour FPGA compiles**
- **System ran for the entire 2009 run-cycle without incident.**

Improvements for 2010 Run Cycle

- **Upgraded cRIO Hardware.**
  - **Faster CPU**
  - **VIRTEX 5 FPGA**
    - **Plenty of FPGA Space**
- **Upgraded from LabVIEW 8.5.1 to LabVIEW 2009**
- **Faster Compile Server**
  - **3 hour compiles (Max)**
- **Full functionality for all Binary Channels**
- **Added Analog Channels (2nd cRIO)**

Current Results (2010)



- **Both Systems Have Been Running for 4 Months Without Incident**

Experience:

- **Noise on Analog Inputs.**
  - **2 Pole Butterworth Filter Applied in FPGA**

- **Our On-Line Reconfiguration Scheme Still Needs Work**

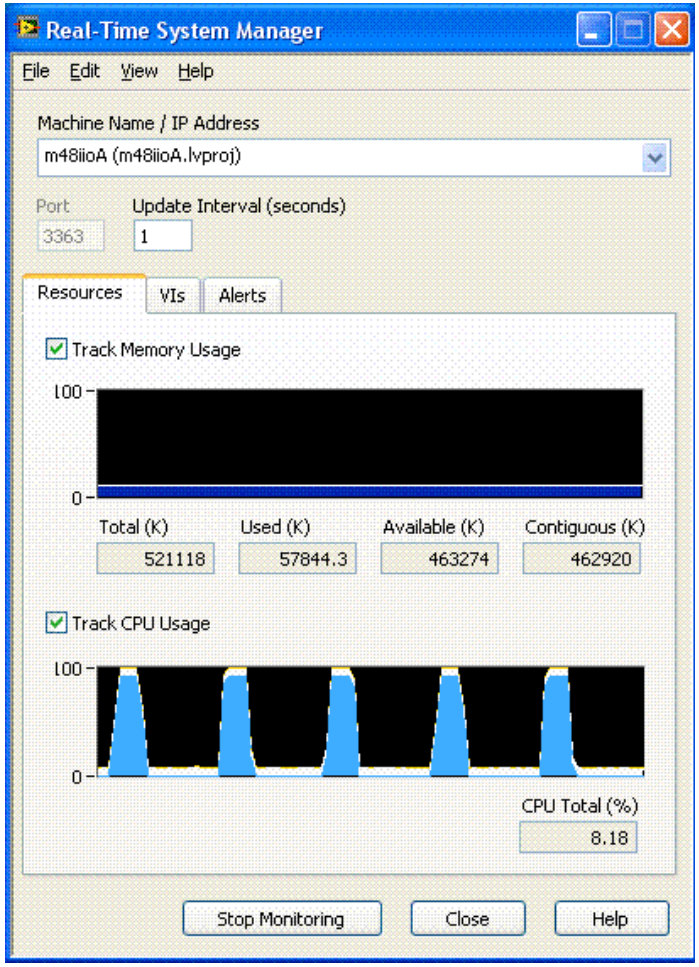
Performance – Benchmarked System

- **Processor**
  - 800 MHz Power PC
  - 512 Mbyte DRAM
  - 4 GB Internal Storage
- **FPGA**
  - LX85 Virtex-5
- **EPICS Load**
  - 57 Records
  - 19 CA Clients
  - 94 CA Connections

Performance – CPU Load

- **iocStats:**
  - **0.18 %**
- **vxWorks "spy"**
  - **317,796,426 %***
- **LabVIEW System Monitor:**
  - **8.18 %**

*Lacks some credibility

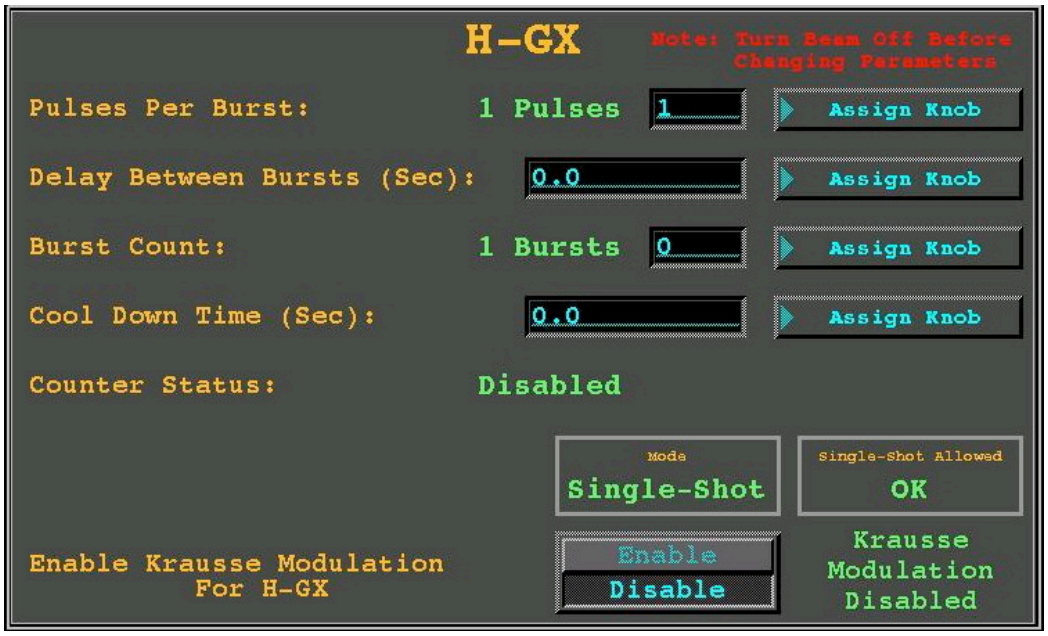


Performance –
Channel Access Benchmarks (using catime)

Test	mv167 (33 Mhz) Items/Sec	Mbps	Speed Increase	cRIO 9024 (800 Mhz) Items/Sec	Mbps	Speed Increase	mv6100 (1.3 GHz) Items/Sec	Mbps	Speed Increase
Channel Connect	1,661.8	0.9	1	5,303.1	2.9	3.19	5,183.7	2.9	3.12
Async Put	10,776.3	1.5	1	26,325.5	3.6	2.44	46,524.5	6.3	4.32
Async Get	9,683.4	2.6	1	22,773.1	6.0	2.35	23,392.7	6.2	2.42
Synch Get	773.9	0.2	1	1,491.4	0.4	1.93	2,569.4	0.7	3.32
Average Perf.			1			1.98			2.63

Future Plans (Immediate)

- **A third embedded EPICS cRIO was installed last month for timing sequence control.**



Future Plans

- **2011 Run Cycle**
  - **Replace Industrial Controls for One Sector (5 modules) with cRIO.**
  - **Replace One Serial CAMAC Loop (4 crates) with cRIO.**
- **cRIO Wire Scanners**
  - **cRIO Event Receiver**
- **More General-Purpose Shared Memory Interface**
  - **Continuation of CosyLAB's Initial Work.**